

## Практическое занятие № 11

Тема: составление программ с функциями в IDE PyCharm Community.

Цели практического занятия: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с текстовыми файлами в IDE PyCharm Community.

Вариант 9

Задача №1:

1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные: Количество элементов:

Индекс последнего максимального элемента:

Меняем местами первую и последнюю трети:

Код:

```
import random

# Генерируем последовательность положительных и отрицательных целых чисел
sequence = [random.randint(-100, 100) for _ in range(20)]

# Записываем последовательность в текстовый файл
with open('original_data.txt', 'w') as file:
    for num in sequence:
        file.write(str(num) + '\n')

print('Исходная последовательность сохранена в original_data.txt')

# Находим индекс последнего максимального элемента
max_element = max(sequence)
max_index = len(sequence) - 1 - sequence[::-1].index(max_element)

# Меняем местами первую и последнюю треть последовательности
third = len(sequence) // 3
new_sequence = sequence[2 * third:] + sequence[third:2 * third] + sequence[:third]

# Записываем обработанную последовательность в новый текстовый файл
with open('processed_data.txt', 'w') as file:
    file.write(f'Исходные данные:\n{sequence}\n\n')
    file.write(f'Количество элементов: {len(sequence)}\n')
    file.write(f'Индекс последнего максимального элемента: {max_index}\n')
    file.write(f'Меняем местами первую и последнюю треть:\n{new_sequence}')

print('Обработанная последовательность сохранена в processed_data.txt')
```

### **Протокол выполнения программы:**

1. Генерация последовательности из 20 случайных целых чисел (от -100 до 100) и сохраняется в файл `original\_data.txt`.
2. Находится индекс последнего максимального элемента в последовательности.
3. Меняются места первой и третьей по последовательности.
4. Сохраняется результат (исходную последовательность, количество элементов, индекс максимального элемента, измененную последовательность) в файл `processed\_data.txt`.

Программа завершает выполнение после сохранения результатов в файлах `original\_data.txt` и `processed\_data.txt`.

### **Задача №2:**

Из предложенного текстового файла (text18-9.txt) вывести на экран его содержимое, количество букв в нижнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку фразой введенной пользователем.

Код:

```
# Чтение содержимого из файла text18-9.txt и подсчёт количества букв в нижнем регистре
with open('text18-9.txt', 'r', encoding='utf-8') as file:
    content = file.read()
    lowercase_letters_count = sum(1 for char in content if char.islower())

print('Содержимое файла text18-9.txt:')
print(content)
print(f'Количество букв в нижнем регистре: {lowercase_letters_count}')

# Введите последнюю строку для добавления в стихотворную форму
last_line = input('Введите последнюю строку для стихотворения: ')

# Запись текста в стихотворной форме в новый файл poem.txt
with open('poem.txt', 'w', encoding='utf-8') as file:
    # Разбиваем текст на строки
    lines = content.splitlines()

    # Формируем стихотворение, добавляя последнюю строку
    poem_lines = lines[:-1] # Берём все строки кроме последней
    poem_lines.append(last_line) # Добавляем введенную пользователем последнюю строку

    # Записываем стихотворение в файл
    file.write('\n'.join(poem_lines))

print('Строки из файла text18-9.txt были оформлены в стихотворной форме в файл poem.txt.')
```

### **Протокол выполнения программы:**

1. Открывает файл `text18-9.txt` для чтения текстового значения с использованием кодировки `utf-8`.
2. Считываем стандартный файл и подсчитываем количество букв в хранилище.
3. Выводим обязательный файл `text18-9.txt` на экран.
4. Выводим количество букв в запись из файла.

5. Просим пользователя ввести текстовый текст для добавления стихотворной формы.
6. Открывает файл `роет.txt` для записи текста в стихотворной форме с использованием кодировки `utf-8`.
7. Разбиваем текст на строки.
8. Формируемое стихотворение, оставляющее все строки, кроме нижней, и добавляющую указанную с помощью текста форму.
9. Записываем сформированное стихотворение в файл `роет.txt`.
10. Выводим информацию о завершении программы, сообщая о том, что строки из файла `text18-9.txt` были оформлены в стихотворной форме в файл `роет.txt`.

**Вывод:** я закрепил знания, понятия, алгоритмы, основные принципы составления программ и приобрел навыки составления программ с текстовыми файлами в IDE PyCharm Community.

